Volume 4, Issue 10, October 2014 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

A Proportional Analysis on Mobile Agent and Modified Tabu Search Algorithm for Routing in Wireless Adhoc Networks

V. Jayaraj[1], J. Jegathesh Amalraj[*2], L. Nagarajan[3], U. Durai[4]

[1,2,3]Bharathidasan University

[1,2,3]Tiruchirappalli, Tamilnadu, India

[4]Periyar University, Salem, Tamilnadu, India

***Abstract** — This Effective transmission of information is done by using efficient routing. The quality constraints have increased due to the variety of applications involved. This has paved the way for quality aware routing. While in the case of wireless networks, the requirements add up more, because it becomes mandatory that we maintain an updated network structure for efficient routing. This paper provides various methods that perform efficient topology discovery and route formulation, along with route maintenance. A new approach called Modified tabu search is proposed and compared with existing Mobile Agent approach. Both the methods are compared with each other by using the parameters such as drop rate, velocity (m/s), Packet Delivery Ratio and Total Over head. The experimental analysis of both the techniques implies that the proposed techniques overtook the Mobile Agent in all the terms.*

***Keywords** — Mobile Agents; Topology discovery; Neighbour Discovery: Tabu Search; Routing; Clustering; SENCAST*

I. INTRODUCTION

Wireless ad hoc networks have increased recently due to the availability of technologies that facilitate interconnectivity. But the problem arises due to the intrinsic properties of wireless networks, mobility. This property of a wireless node makes the topology discovery of the network a difficult task. There exist various topology discovery techniques designed specifically for wireless networks. The topology discovery in a wireless network is basically a neighbour discovery mechanism that finds all nodes within the range of the current node and marks them as neighbours. Wireless networks can be congregated into two categories: infrastructured networks; and infrastructure-less networks[18]. Infra structured network is with fixed and wired gateways known as Access points. The second type is also known as ad-hoc networks. It consists of a collection of geographically distributed nodes that communicate with one another over a wireless medium without the need of fixed routers [19]. The motivation for adhoc-networks is military applications.

II. LITERATURE REVIEW

[1] presents a neighbor discovery algorithm called MultiPath OLSR (MP-OLSR), is a multipath routing protocol based on OLSR [2]. The Multipath Dijkstra Algorithm is proposed to obtain multiple paths. The algorithm gains great flexibility and extensibility by employing different link metrics and cost functions. It performs the process of route recovery and loop detection, along with improving the quality of service regarding OLSR. It is also backward compatible with OLSR. [3] presents a reliable neighbourhood discovery algorithm. It proposes two algorithms, and also provides process guarantees. The initial algorithm is a region based neighbor discovery algorithm, which does not guarantee communication links. Another algorithm is proposed, which integrated the reliability into the previous algorithm.

The process of routing is one of the very important components when considering data transmission. Routing is the process which determines the available routes from a source node to a destination node. It is always optimal to use a routing algorithm that satisfies the quality constraints along with an optimal path. [4] provides an extensive survey on routing protocols and their categories. It divides the wireless routing protocols into proactive or reactive, and geographical, multipath, hierarchical, flow oriented, WMN, multicast, geo-cast, power aware and hybrid. [5] provides a comparative analysis on AODV, DSR, OLSR and DSDV protocols. All simulations were conducted on NS2, and the results provide a valuable insight on the working of these protocols. [6] presents a neighbor supported reliable multipath multicast routing in ad-hoc networks. A mesh of multipath routes are established from source to multicast destinations using neighbors that have high reliability pair factor. MMRNS operates in the following phases. (1) Reliability pair factor is determined based on node power level. (2) Pruning neighbour nodes that have reliability pair factor smaller than a threshold. (3) Discovery of multipath multicast mesh routes with the help of request and reply packets. (4) Multipath priority assignment based on minimum value of reliability pair factor of a path and information transfer from source to the multicast destinations and (5) route maintenance against link/node failures.

The process of clustering has been used in networks for grouping up of nodes and making them function as a single entity rather than independent entities. A cluster leader is to be elected for each cluster to facilitate communication within a cluster and between clusters. [7] describes a fast converging clustering approach on MANET. [8] presents a graph clustering approach for MANETs. It describes a new graph clustering algorithm that automatically defines the number of

© 2014, IJARCSSE All Rights Reserved

clusters based on a clustering tendency connectivity-based validation measure. It describes three phases, the coarsening phase, partition phase and refinement phase for performing the clustering process. [9] presents an efficient graph based clustering algorithm.
In [10], a dynamic algorithm that performs unicasting and multicasting is used. The SENCAST [10] helps in efficient data transfer. SENCAST is used in our paper for the estimation of the bandwidth in the routes, that serves as a QoS parameter when determining the routes. [17] provides an efficient algorithms that utilizes an existing cognitive network to exhibit better performance. This methodology also ensures that the existing channel frequencies are not wasted and the frequencies that are in use are utilized to the maximum extent without the need for bothering about the effects of interference.

III. MOBILE AGENTS

A mobile agent based system that allows for real time topology discovery of the network. As opposed to many such mobile agent based systems that utilize "Ant" based algorithms, our system makes efficient use of the limited network resources of the ad hoc network by restricting the mobility of the agents. The mobile agent uses its mobility only to move to a new node after which it turns into a stationary agent. Mobile agents are agents that can physically travel across a network, and perform tasks on machines that provide agent hosting capability. This allows processes to migrate from computer to computer, or processes to split into multiple instances that execute on different machines, and to return to their point of origin. Unlike remote procedure calls, where a process invokes procedures of a remote host, process migration allows executable code to travel and interact with databases, file systems, information services and other agents. Mobile agents server as a very practical solution for topology discovery in an environment where the topology changes very frequently [20]. Due to the decentralized characteristics of ad hoc network, it is essential that the topology discovery system is not centralized like traditional topology discovery system. In addition to being decentralized the topology discovery system also has to be "intelligent".

```
<?xml version='1.0' encoding='UTF-8' ?>
- <tXML generator='1' fileID='2006-02-28T11:00:26' timestamp='2006-02-28T11:00:26-08:00'>
    <link id='0' node1='1' node2='2' />
    <link id='2' node1='1' node2='3' />
    <link id='3' node1='2' node2='4' />
    <link id='4' node1='4' node2='5' />
  </tXML>
```


Fig 1: Topology Information XML

The Fig 1 shows how the topology information in passed in XML format. This allows the agent to query the xml file in an efficient way rather than parsing through plain text file.
The platform that an agent originates is called home platform and is assumed as a trusted environment for that agent.

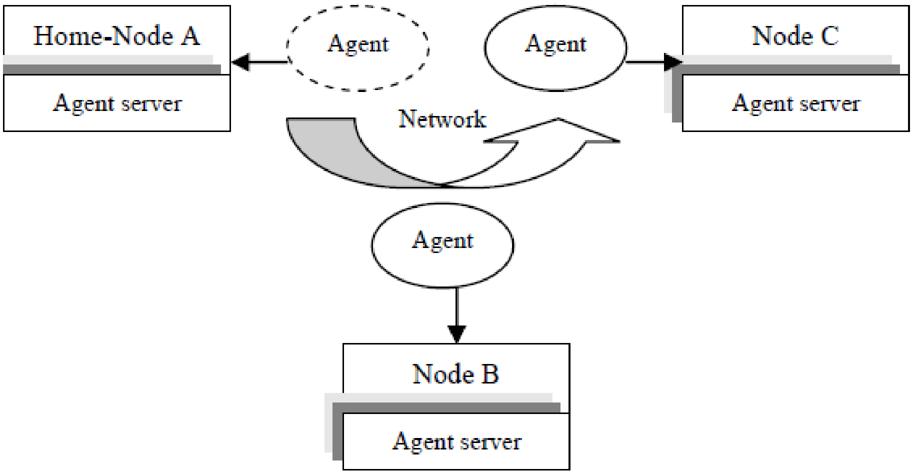

Fig 2: Mobile agent system

The Fig 2 shows a mobile agent system. Each mobile node will run an agent server that provides the basic functionality for static and mobile agents, such as migration, communication, and security. Possibly, the agent server will be written in Java language due to its object-orientation nature, object serialization and remote method invocation techniques, and enhanced security. Static agents will be resident on mobile hosts and will be continuously running. These agents will be mainly responsible for the following operations: maintain a routing table; decide the best path to route network traffic based on information found on the routing table; and monitoring system's resources in terms of memory capacity, processing capabilities, network

© 2014, IJARCSSE All Rights Reserved

performance, cost, and so on. On the other hand, mobile agents will be responsible for the following operations: collect information generated from static agents; update routing tables on mobile hosts; and discovering new routes. They must also inform static agents; and other mobile agents for changes in the network. Static agents will maintain routing tables continuously, decide the best path dynamically, and monitor system's resources periodically. Mobile agents will collect information from static agents periodically, update routing tables periodically, and communicate with static agents and other mobile agents when necessary. Figure 3 illustrates the model with four mobile hosts, four static agents, and two mobile agents.

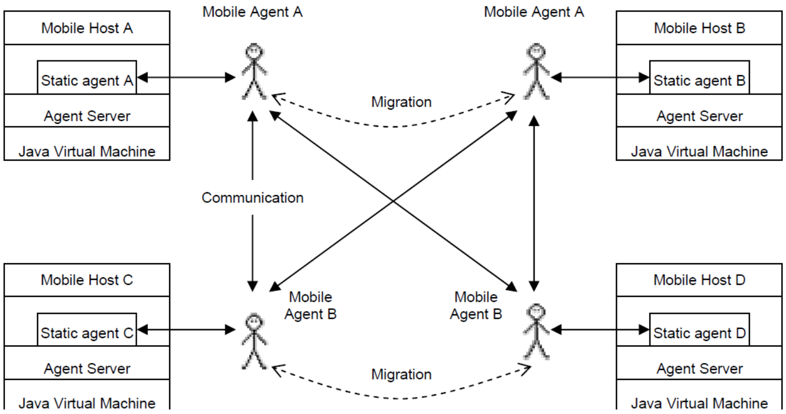

Fig 3: The proposed model

IV. MODIFIED-TABU SEARCH

The basis for any routing mechanism is the awareness in the network topology. Our process begins by an efficient topology discovery mechanism followed by the route formulation phase that helps in discovering the available routes and the route optimization phase that is performed by using the modified Tabu search. Further, an error handling routine is also incorporated, that troubleshoots the obtained routes for any disruptions in the network.

A. Topology Discovery

The topology discovery is the process of determining the nodes that are present in the network and their connections with other nodes. Along with the topology discovery, clustering and cluster head selection process is also carried out to ensure efficient transfer mechanism. Further, the topology refresh is also carried out in order to make sure that the topology that has been discovered remains latest and without any dead paths.

1) Neighbor Discovery

The neighbor discovery is performed by finding the one hop neighbors. Due to the fact that a wireless ad-hoc network is used, we cannot be sure that the network topology remains the same, hence, in case of emergency data transfers, the next node (neighbor) in range can be used. The process of neighbor discovery is performed by using the '*hello'* messages [16]. Each node broadcasts the *hello* message hence retrieves details about all other nodes in the network that are within its range.

2) Clustering

Clustering is the process of grouping similar data, such that all nodes that are similar to each other occupy the same cluster, and those that have differences occupy different clusters. A graph based Clustering that determines the clusters along with the cluster leaders is proposed here. Consider a graph *G=(V,E)*, with vertices V and edges E. The number of nodes in the graph G is *n*, and the edge set contains *(i,j)* where i and j represent vertices from V. The weight matrix is represented by $W_{ij,}$ that represent the cost of traversal from i to j, here it represents the distance from node i to node j.

$$w_{ij} = \begin{cases} distance\ between\ i\ and\ j\ \ if\ \ a\ \ path\ exist \\ 0 \qquad\qquad\qquad\qquad otherwise \end{cases} \qquad \ldots (1)$$

The adjacency matrix A_{ij} is created, that represents the connectivity in the network. Adjacency matrix is determined by the following

$$A_{ij} = \begin{cases} 1\ if\ there\ is\ a\ connection\ between\ i\ and\ j \\ 1 \qquad\qquad if\ i=j \\ 0 \qquad\qquad otherwise \end{cases} \qquad \ldots (2)$$

The degree of a node Deg_i, determined by the adjacency matrix, is the number of connections that a node maintains in the network.

$$Deg_i = \sum_{j=1}^{n} aij - 1 \qquad \ldots (3)$$

In a wireless network, clustering is performed by dividing the network into partitions such that the distance constraint is satisfied. A node is added to the cluster, if it contains the threshold that is less than that of the defined threshold. The value of threshold [9] is defined by,

$$\delta = D_{\min} + (D_{\max} - D_{\min}) \times cons\tan t \qquad \ldots (4)$$

Where D_{min} and D_{max} defines the minimum and maximum distances exhibited between nodes in the network, and the *constraint* defines the accuracy of the cluster. The value of a constraint ranges from 0 to 1. A very low constraint groups only closely grouped nodes and many clusters, while a high value provides clusters of large size and lower cluster count. It is very important to strike a balance in this value. There is no defined rule to identify this value, this can only be performed as a trial and error process, and it differs with networks.

Topology Discovery Algorithm

```
For each node in v {
  Select all nodes (x) with Deg_n > 0
  For each x,
  Find threshold δ_x using (equn no for Del
      If δ < thresh_max
      Add it to cluster C_n.
  End For
End For
```

For every node n in the vertex set V, its neighbours are determined. Every neighbour node is processed to find the threshold using 1. If the threshold is within the limits of the defined threshold $thresh_{max}$, then the node is incorporated in the current cluster.

Cluster Determination Algorithm

```
For each cluster C_i
   For each Cluster c_j where j>i
     Check if all elements in C_i ⊂ C_j
     If yes delete C_i
  End For
End For
```

The algorithm for cluster finalization is described above. Each cluster is examined to see if the cluster elements are a subset of any other cluster. If so, the current cluster is removed from the cluster set. The clusters remaining after this process forms the final clusters and the cluster base values are assigned as the cluster leaders.

3) Leader Selection

In general, when considering the process of routing, one cluster leader alone is not sufficient to maintain a stable cluster [12]. The following properties describe the characteristics of a cluster node and a leader node for maintaining the stability.

Property 1: *All nodes in a cluster should contain 2f+1 neighbors and f+1 disjoint routes to the leaders.*

Property 2: *If a leader connects to less than 2f+1 nodes, then the cluster is taken apart.*

where *f* denotes the threshold for the occurrence of an anomaly.

4) QoS Parameter Maintenance

Due to the fact that the system also considers the QoS parameters during the transmission of data, the process of topology discovery performs an extra job of obtaining the QoS parameters of the neighbors during the neighbor discovery phase. Since we select the routes in the network on the basis of both availability and the quality parameters, it becomes necessary for the network to maintain these parameters.

5) Topology Refresh Phase

It cannot be expected of a wireless network to remain stable [15]. Hence it becomes vital to be certain the routes determined earlier are live and working. This functionality is performed by the topology refresh phase. This module checks for the validity of the network after a particular determined time interval. If so, it intimates the rest of the cluster mates about the invalid route and tries to find a new route. This functionality is performed by all the leaders. The leaders take turns in performing this operation. Hence, we can be positive that at a certain time, only one of the leader's is performing the refresh operation. Hence the network traffic created due to the process of refresh is maintained minimal.

B. Route Formulation

A process requesting for data transfer requires a path for transmission of the messages. The process of finding the optimal path in which the data transfer can take place is referred to as routing. Our process of route formulation uses the available routes and the QoS parameters to determine a set of routes. The optimal route from these set of routes is determined by using the modified Tabu Search.

Route Formulation Algorithm

RREQ:
If(sender)
    Prepare RREQ packet
    Transmit to all connected leaders

Leader:
Search for the destination candidate
  If available
    Convert packet to RREP
    Append current node info
    Transmit base to the Previous node
  Else
    Append current info and transmit to connected leaders

RREP:
If (sender)
  Reverse map the RREP packet to determine the route
Else
  Transfer packet to prev node in the node list

1) Multiple Route Selection

When a routing need arises, the source node sends a Route REQuest (RREQ) packet [11] to all its leaders. The leaders determine if they have any routes already present for the source and destination in the Tabu Queue. If such a route exists, the route is added to the *Routes* list. The request is then transmitted to all the other leader nodes, along with the current node list embedded in the RREQ message. Every leader node transmits the information available within their cluster. This process returns all the best available routes [13] back to the base node. The base node performs reverse mapping of the received packet and determines all the available routes [1] and populates them in the *Routes* list.

2) Route Shortlist

The process of bandwidth estimation is used as a QoS parameter to shortlist the available routes from the *Routes* list [14]. The bandwidth estimation is performed using SENCAST, a dynamic routing protocol. The QoS requirement of the required application is also taken into account and the shortlisting of the routes is performed. This process returns a set of shortlisted routes *RouteShortlist*. This list also contains a set of routes. The optimal service is determined by using the Modified Tabu Search algorithm.

3) Route Optimization using Modified Tabu Search

A modified form of Tabu search is used to determine the final optimal route. This method uses the intermediate term memory, which . a random route RR_i is selected from the *RouteShortlist.* The cost of transmission and level of QoS support is determined and is provided a score $sBest_i$. Another random route RR_j is selected from the set, where $RRi \neq RRj$. The $sBest_j$ is calculated and is compared with the $sBest_i$. The least cost *sBest* is considered for transmission, and this route is added to the Tabu list.

Modified Tabu Search Algorithm

Select a random route RR_i
  Find cost incurred. Let it be sBest
  Select another random route $RR_,$ where $i \neq j$
  If cost (RR_j) < cost (RR_i)
    sBest = RR_j
    Use sBest to transfer data
    Add sBest to Tabu Queue

4) Route Maintenance

The process of route maintenance plays a vital role in maintaining the stability of the system. The process of route maintenance begins from the phase when the first route is selected. A wireless ad hoc network in general is dynamic. i.e. nodes join and leave the network, and moves from one location to other in a random manner. The random joining and leaving phase is handled by the topology refresh phase, while node movements are handled by this module.

When a node *N* wishes to move from its current position, it broadcasts a *leave* message to all its connected node (and leaders) intimating movement. All the nodes receiving the message perform the process of deleting *N* and all the routes concerning *N* and transfer the details to all the leaders connected to it. When a node enters a region through migration, it broadcasts a *join* message to intimate the surrounding nodes that it has entered into their vicinity. This process will require some addition in the neighbor data of the candidate nodes and the leaders. All connected leaders are also intimated about this newly joined node.

Sometimes, the migration process might not have any effect on some nodes. This occurs when a node moves, but still under the vicinity of some of its previous peers or leaders. Hence this process might trigger an unnecessary deletion and addition of data. Further, there is also a high probability of loss of some routing data. Hence all nodes wait for a threshold time interval *RefreshWait* before adding or deleting any nodes. This provides an added advantage of facilitating multiple additions or deletions, since we can also add up the transactions that occurred during the wait time.

© 2014, IJARCSSE All Rights Reserved

Route Maintenance Algorithm

```
If (movement triggered)
  Obtain neighbor set N
  Multicast leave message (leave_n) to N

If (Movement Stopped)
  Broadcast join_n msg

If(msg received)
  Add to node list Q_n
  Trigger RefreshTimer using refreshWait value
If (RefreshTimer  expired)
  For all msgs in list Q_n
    If( leave_n)
      Check for join_n
      If (not available)
        Remove n from neighbor list
      Else
        Remove join_n from Q_n
        Remove leave_n from Q_n
    If (join_n)
      Add n to neighbor list
      Remove join_n from Q_n
  End For
```

V. RESULTS AND DISCUSSIONS

Both the Mobile Agent and Modified Tabu Search Algorithm is implemented in the NS2 environment to evaluate both the performance. It is observed the performance of Modified Tabu Search Algorithm is better than Mobile Agent in the case of Drop Rate and time taken to find the route between the node A to B. The Figure 12, and 13 symbolizes the performance of Modified Tabu Search over Mobile Agent.

The Figure 12 compares the performance of Modified Tabu Search with Mobile Agent based on drop rate. It is observed that the Mobile Agent has higher drop rate than the Modified Tabu Search. As the number of packets increases the drop rate is increasing in the case of Mobile Agent, where as promising results on drop rate is observed in the case of Modified Tabu Search.

The Figure 13 compares the performance of the time taken to find the best route from the node A to node B. The Figure symbolizes the Modified Tabu Search over performs than the Mobile Agent. The list of nodes and topology is stored in the tabu search list. So in the case of Modified Tabu Search, the node may refer the list for further assistance, which will reduce the time. Again in the case of the loss of node or break down of topology, the refresh phase is possible in Modified Tabu Search. In Mobile Agent, that things are find to be impossible. In every time, the Mobile Agent has to go through every node to determine its topology, which will either increase the time. But in the case of Modified Tabu Search it is found that the time is either found to constant or decreasing which proves the performance of Modified Tabu Search over Mobile Agent.

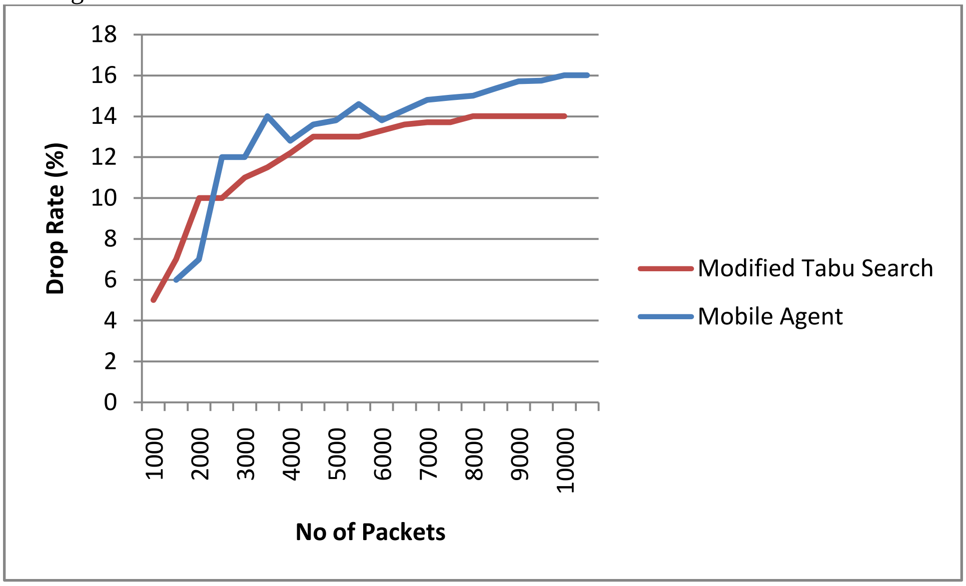

Fig 4: Comparison of Modified Tabu Search with Mobile Agent based on Drop rate

© 2014, IJARCSSE All Rights Reserved

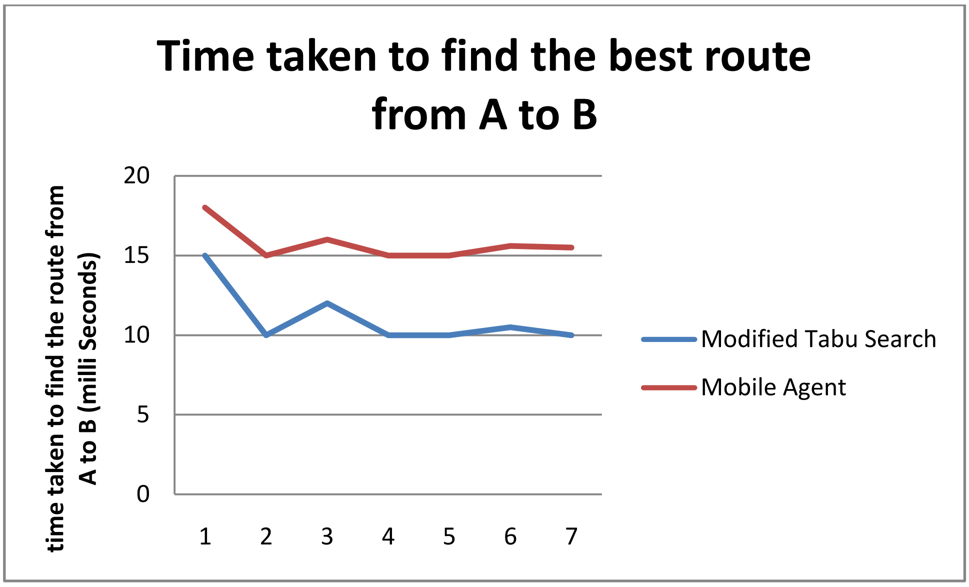

Fig 5: Comparison of the time taken to find the best route from the node A to node B

VI. CONCLUSIONS

Routing in a wireless ad hoc network is a critical process that involves many parameters and security issues. In this paper, two methods such as Mobile Agent and Modified Tabu Search is compared with each other. From the results we can see that the proposed Modified Tabu Search process works efficiently when considering the drop rate. And the threshold for velocity and maximum number of nodes was determined using the packet delivery ratio and the overhead calculation was performed with respect to the cluster size. Efficiency in the results were observed. Hence we can conclude that the above method performs well under all random circumstances in a Adhoc network.

REFERENCES

[1] Jiazi Yi, Asmaa Adnane, Sylvain David and Benoit Parrein, 2011. Multipath Optimized Link State Routing for Mobile Ad Hoc Networks, Ad Hoc Networks, Elsevier, Volume 9, Issue 1, (January 2011), Pages 28–47.

[2] T. Clausen and P. Jacquet 2003. IETF Request for Comments: 3626, Optimized Link State Routing Protocol OLSR, October 2003.

[3] Alejandro Cornejoa, Saira Viqarb and Jennifer L. Welchb, 2014. Reliable Neighbor Discovery for Mobile Ad Hoc Networks, Elsevier, Volume 12, January 2014, Pages 259–277.

[4] Eiman Alotaibi and Biswanath Mukherjee, 2012. A Survey on Routing Algorithms for Wireless Ad-Hoc and Mesh Networks, Computer Networks, Elsevier, Volume 56, Issue 2, 2 February 2012, Pages 940–965.

[5] S. Mohapatraa and P.Kanungob, 2012. Performance analysis of AODV, DSR, OLSR and DSDV Routing Protocols using NS2 Simulator, Procedia Engineering, Elsevier, Volume 30, Pages 69–76.

[6] Rajashekhar C. Biradar and Sunilkumar S. Manvi, 2012. Neighbor Supported Reliable Multipath Multicast Routing in MANETs, Journal of Network and Computer Applications, Volume 35 Issue 3, May, 2012, Pages 1074-1085.

[7] Liren Zhang and Hesham El-Sayed, 2012. A Novel Cluster-Based Protocol for Topology Discovery in Vehicular Ad Hoc Network, Procedia Engineering, Volume 10, Pages 525–534.

[8] Maria C. V. Nascimento and Andre C. P. L. F. Carvalho, 2011. A Graph Clustering Algorithm Based on a Clustering Coefficient for Weighted Graphs, Journal of the Brazilian Computer Society, Springer-Verlag Publisher, March 2011, Volume 17, Issue 1, Pages 19-29.

[9] Zhou Mingqiang, Huang Hui and Wang Qian, 2012. A Graph-based Clustering Algorithm for Anomaly Intrusion Detection, Procedings of 7th International Conference on Computer Science & Education (ICCSE), Pages 1311 – 1314, IEEE.

[10] P. Appavoo and K. Khedo, 2008. SENCAST: A Scalable Protocol for Unicasting and Multicasting in a Large Ad hoc Emergency Network, IJCSNS International Journal of Computer Science and Network Security, VOL.8 No.2, February 2008, Pages 154-165.

[11] Tran The Son, Hoa Le Minh, Graham Sexton and Nauman Aslam, 2014. A Novel Encounter-Based Metric for Mobile Ad-Hoc Networks Routing, Ad Hoc Networks, Elsevier, Volume 14, March 2014, Pages 2–14.

[12] P.M. Mafra, J.S. Fraga and A.O. Santin, 2014. Algorithms for a distributed IDS in MANETs, Journal of Computer and System Sciences, Elsevier, Volume 80, Issue 3, May 2014, Pages 554–570.

[13] Salim Bitam, Abdelhamid Mellouk and Sherali Zeadally, 2013. HyBR: A Hybrid Bio-inspired Bee swarm Routing protocol for safety applications in Vehicular Ad hoc NETworks (VANETs), Journal of Systems Architecture, Elsevier, Volume 59, Issue 10, Part B, November 2013, Pages 953–967.

[14] Szu-Lin Su and Yi-Wen Su, 2011. Single Phase Admission Control for QoS-routing Protocol in Ad Hoc Networks, Journal of Ad Hoc Networks, Volume 9 Issue 7, September, 2011, Pages 1359-1369.

© 2014, IJARCSSE All Rights Reserved

[15] Nityananda Sarma and Sukumar Nandi, 2010. Route Stability Based QoS Routing in Mobile Ad Hoc Networks, Wireless Personal Communications, Springer, July 2010, Volume 54, Issue 1, Pages 203-224.

[16] Giovanni Di Crescenzo, 2007. Secure Node Discovery in Ad-hoc Networks and Applications, Electronic Notes in Theoretical Computer Science, Elsevier, Volume 171, Issue 1, 14 April 2007, Pages 43–55.

[17] Veerasamy Jayaraj and Jegathesh Amalraj J, 2012. “Efficient Spectrum Sharing and Allocation Schemes for Throughput Enhancements in a Cognitive Radio Network”, Procedings of Third International IEEE Conference on Computing, Communication And Networking Technology (ICCCNT’2012), SNS College of Engineering, Coimbatore, Tamilnadu, July 2012, IEEE.

[18] E.M. Royer and C.K Toh, “ A review on current routing protocols for Ad-hoc mobile Wireless networks”, IEEE Personal Communciation, April 1999, pp 46-55.

[19] R. Rajaraman, “ Topology control and Routing in ad-hoc network: A survey”, north Eastern University, Boston 2002, ACM press pp 60-73.

[20] Nikos Migas, William J. Buchanan, Kelvin A. Mcartney, “ Mobile Agent for Routing Topology Discovery and Automatic network Reconfiguration in ad-hoc networks”, 2002.

[21] Romit roy choudary, S. Bandhopadhyay, Krishnan paul, “ Topology discovery in adhoc network by using Mobile Agents”, mobile Agents for Telecommunication Application, Springer 2000, pp 1-15.

© 2014, IJARCSSE All Rights Reserved